

Learning Temporal Alignment Uncertainty for Efficient Event Detection

Iman Abbasnejad^{1,2}, Sridha Sridharan¹, Simon Denman¹, Clinton Fookes¹, Simon Lucey²

¹Image and Video Laboratory, Queensland University of Technology (QUT), Brisbane, QLD, Australia

²The Robotics Institute, Carnegie Mellon University, 5000 Forbes Ave, PA, USA

Email: {i.abbasnejad, s.sridharan, s.denman, c.fookes}@qut.edu.au, slucey@cs.cmu.edu

Abstract—In this paper we tackle the problem of efficient video event detection. We argue that linear detection functions should be preferred in this regard due to their scalability and efficiency during estimation and evaluation. A popular approach in this regard is to represent a sequence using a bag of words (BOW) representation due to its: (i) fixed dimensionality irrespective of the sequence length, and (ii) its ability to compactly model the statistics in the sequence. A drawback to the BOW representation, however, is the intrinsic destruction of the temporal ordering information. In this paper we propose a new representation that leverages the uncertainty in relative temporal alignments between pairs of sequences while not destroying temporal ordering. Our representation, like BOW, is of a fixed dimensionality making it easily integrated with a linear detection function. Extensive experiments on CK+, 6DMG, and UvA-NEMO databases show significant performance improvements across both isolated and continuous event detection tasks.

I. INTRODUCTION

A popular strategy for learning a discriminative event detection function, $f(\mathbf{X}; \theta) : \mathbb{R}^{D \times M} \rightarrow \mathbb{R}^1$, is to employ a linear function,

$$f(\mathbf{X}; \theta) = \phi\{\mathbf{X}\}^T \theta \quad (1)$$

where $\phi\{\mathbf{X}\}$ is a vectorized feature representation of the multi-dimensional event sequence $\mathbf{X} \in \mathbb{R}^{D \times M}$; D is the dimensionality of the signal; and M is the number of frames. This is in contrast to canonical methods for temporal detection in vision such as hidden Markov models (HMMs) [1], latent dynamic conditional random fields (LDCRFs) [2], time series kernels [3], [4] and dynamic time-alignment kernels [5] which have non-linear interactions between the model parameters, θ , and the feature representation, $\phi\{\mathbf{X}\}$.

There are two central advantages for maintaining a linear relationship between $\phi\{\mathbf{X}\}$ and θ in Equation 1. Firstly, the linear form allows one to employ canonical max-margin linear detectors such as linear support vector machines (SVM) [6] or structural output SVMs (SO-SVM) [7] which generalize well to high-dimensional discriminative learning problems. Secondly, during detector evaluation one can take advantage of efficient search strategies afforded to linear detectors (i.e. linear convolution, summed area tables, etc.) making the application of such detectors highly efficient.

Recently, [8], [9] demonstrated that state-of-the-art performance in temporal event detection can be achieved using a *bag of words* (BOW) representation of the temporal signal in conjunction with a SVM-style detector. Specifically, the authors

compared their approach to canonical hidden state probabilistic methods for event detection such as hidden Markov models (HMMs), and demonstrated their BOW+SVM method achieves superior performance in terms of computation and accuracy by a considerable margin. A drawback, however, to the BOW representation lies in the destruction of the temporal dynamics in the raw signal, \mathbf{X} . It is the preservation of this temporal ordering information that is at the heart of this paper.

Contributions: We make the following contributions in this paper,

- We propose a novel strategy for learning the relative alignment uncertainty between pairs of training sequences using an adaptation of dynamic time warping (DTW). Using this model of uncertainty we then propose a new representation which is an efficient linear transform of the raw input sequence which: (i) preserves temporal ordering information while averaging over alignment uncertainty, and (ii) ensures the representation is of a fixed dimensionality so as to be applicable within a linear event detection function.
- We demonstrate that our approach has comparable computational cost to current state-of-the-art BOW linear detectors, but with the advantage of obtaining significantly better detection performance across the CK+, 6DMG, and UvA-NEMO event detection datasets.

We evaluate the proposed approach on three datasets for both isolated and continuous event detection, and demonstrate improved performance while retaining computational efficiency. The remainder of this paper is structured as follows: Section II presents an overview of existing literature, in particular the bag of words representation, dynamic time warping and time series kernels; Section III presents our proposed approach and in Section IV we outline the features that we use in the proposed method; Section V evaluates our proposed approach; and Section VI concludes the paper.

II. BACKGROUND

A. Bag of Words Representation

Bag of words (BOW) representations can be viewed as simply taking the mean over all frames of a non-linear representation $\eta\{\mathbf{x}_m\}$, where the m -th frame vector is $\mathbf{X} =$

$[\mathbf{x}_1, \dots, \mathbf{x}_M]$, such that,

$$\phi\{\mathbf{X}\} = \frac{1}{M} \sum_{m=1}^M \eta\{\mathbf{x}_m\}. \quad (2)$$

The non-linear function obtains a sparse encoding of the frame vector, \mathbf{x} , using the codebook matrix $\mathbf{D} \in \mathbb{R}^{D \times K}$, where K is the number of codebook entries. The codebook is typically learned through k-means clustering. We can define this non-linear function as,

$$\eta\{\mathbf{x}\} = \arg \min_{\mathbf{b}} \|\mathbf{x} - \mathbf{D}\mathbf{b}\|, \quad (3)$$

s.t. $\mathbf{b} \in \mathbb{B}$

where $\mathbb{B} = \{\mathbf{e}_k\}_{k=1}^K$ is the non-convex set of all K dimensional vectors, \mathbf{e}_k , containing all zeros except for one at the k -th entry.

An initial question one may ask is why destroy the temporal ordering information in \mathbf{X} ? One obvious motivation stems from the realization that the vectorized dimensionality of \mathbf{X} will vary as a function of M , whereas $\phi\{\mathbf{X}\}$ is invariant to M . The fixed dimensionality of $\phi\{\mathbf{X}\}$ allows for training with canonical linear geometric classifiers such as linear SVM and structural output SVM. The inevitable information loss stemming from the taking the multi-dimensional average over all frames is somewhat mitigated by the application of the non-linear mapping in Equation 3. Without the non-linear mapping, one would simply be learning a detector model θ from the multi-dimensional mean of \mathbf{X} across frames. By encoding \mathbf{X} non-linearly the destruction of information is not quite as severe with higher-order statistical moments being preserved (i.e. $\phi\{\mathbf{X}\}$ can be interpreted as a multidimensional histogram feature).

Cost of Search: Another advantage of the BOW representation is that since temporal ordering information is destroyed in Equation 2, searching over variable size window widths becomes computationally efficient through the judicious use of a summed area table (commonly referred to as the integral image [10] in computer vision). In this strategy once we have applied the non-linear transform in Equation 3 to all frames in a sequence, one can then obtain a cumulative sum of the sequence, at a cost of $\mathcal{O}(MK)$, and then obtain the BOW representation for any sub-window at a cost of only $\mathcal{O}(K)$ operations. The sum area table method can only be employed for sequence representations such as BOW where temporal ordering is destroyed. The major computational drawback to the BOW representation is the cost of mapping from $\mathbf{X} \rightarrow \eta\{\mathbf{X}\}$ is $\mathcal{O}(MKD)$ using a naive codebook search.

B. Dynamic Time Warping

A number of works have been proposed in the literature for temporal alignment [11], [12]. In this work we use dynamic time warping (DTW) due to its established performance on temporal alignment tasks.

Lets assume we have two multi-dimensional sequences, $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_M]$ and $\mathbf{Y} = [\mathbf{y}_1, \dots, \mathbf{y}_N]$, of equal dimensionality D (i.e. $\mathbf{x} \in \mathbb{R}^D$ and $\mathbf{y} \in \mathbb{R}^D$) but differing frame lengths, M and N respectively. We would like to temporally align these two sequences based on some

distance metric. For our purposes this will be the Euclidean distance. Dynamic time warping (DTW) can be applied to align the two signals, and this can be expressed as solving,

$$\text{DTW}(\mathbf{X}, \mathbf{Y}) = \min_{\pi_x, \pi_y} \sum_{t=1}^T \|\mathbf{X}[\pi_x(t)] - \mathbf{Y}[\pi_y(t)]\|_2^2 \quad (4)$$

where π_x and π_y are integer index vectors with the constraints that $1 = \pi_x(1) \leq \pi_x(2), \dots, \leq \pi_x(T-1) \leq \pi_x(T) = M$ and $1 = \pi_y(1) \leq \pi_y(2), \dots, \leq \pi_y(T-1) \leq \pi_y(T) = N$ with unitary increments and no simultaneous repetitions. The length T of the index vectors π_x and π_y are bound by $T \leq M + N - 1$. For all elements of π_x and π_y we define the increment τ such that

$$\tau = \begin{bmatrix} \pi_x(p+1) \\ \pi_y(p+1) \end{bmatrix} - \begin{bmatrix} \pi_x(p) \\ \pi_y(p) \end{bmatrix} \quad (5)$$

is constrained to the a set of 3 causal moves \rightarrow, \uparrow and \nearrow ,

$$\tau \in \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \end{bmatrix}. \quad (6)$$

It is the constraint of the causal moves defined in Equation 6 that makes an efficient solution to the DTW objective in Equation 4 possible. Specifically, the causal constraints imply a tree-structure which can be solved efficiently through belief propagation (i.e. Viterbi decoding) with a cost of $\mathcal{O}(MND)$.

DTW Warping Matrices: One can re-write the objective in Equation 4 as,

$$\text{DTW}(\mathbf{X}, \mathbf{Y}) = \min_{\mathbf{P}_x, \mathbf{P}_y \in \mathbb{P}} \|\mathbf{X}\mathbf{P}_x - \mathbf{Y}\mathbf{P}_y\|_F^2 \quad (7)$$

where \mathbf{P}_x and \mathbf{P}_y are the $M \times T$ and $N \times T$ warping matrices respectively stemming from the set \mathbb{P} that enforce causal deformations in time. Although unconventional, the concept of expressing the warps stemming from DTW alignment as deformation matrices is crucial later for our proposed approach. Figure 1 shows four different examples with their corresponding alignment paths.

C. Time Series Kernels

Time series kernels have been gaining in popularity recently for temporal classification and event detection [3], [4]. Recently, Lőrincz et al. [4] proposed the idea of employing a kernel SVM based on a time series kernel for event detection. In this approach they proposed an event detection function as,

$$f(\mathbf{X}; \theta) = \sum_{l=1}^L \alpha_l k(\mathbf{X}, \mathbf{X}_l) \quad (8)$$

where $\theta = \{\alpha_l, \mathbf{X}_l\}_{l=1}^L$ are the kernel SVM's model parameters specifically the L support weights α_l (which have the binary support labels subsumed within them) and support vectors \mathbf{X}_l . The alignment kernel is defined as,

$$k(\mathbf{X}_i, \mathbf{X}_j) = \exp\{-t \cdot \hat{\text{DTW}}(\mathbf{X}_i, \mathbf{X}_j)\} \quad (9)$$

where t is a constant. The measure $\text{DTW}()$ in Equation 7 is not technically a distance (as it does not obey the triangle inequality) so the authors propose projecting the result into the closest symmetric positive semi-definite kernel $\hat{\text{DTW}}()$. Lőrincz et al. also proposed various extensions and variations

to the DTW kernel, such as the Global Alignment (GA) kernel, the details of which are outside the scope and focus of this paper.

A real strength of this method is that it elegantly embraces the idea that alignment is a relative notion. Instead of trying to align all sequences to a single temporal frame of reference, the approach instead employs the notion of relative alignment between pairs of training examples. The authors reported state of the art event detection performance across a number of event detection datasets, also validating the importance of preserving temporal ordering information in any representation one employs for event detection.

Computational Cost: Although achieving impressive empirical performance, time series kernels cost $\mathcal{O}(LMND)$ for every window searched in a sequence. L is the number of support vectors, M is the length of the input sequence, N is the average length of the support vector sequences and D is the dimensionality of the sequences. Some work [3] has explored strategies for making these methods more efficient such as the employment of constrained DTWs [13], which consider a smaller set of possible causal alignments. Even with these speed ups, the cost of evaluation is dramatically larger than most other event detection methods in current literature such as efficient BOW methods. Their strength, however, lies in their good empirical performance and the theoretical insight that temporal ordering is of high importance in event detection, and relative DTW alignment may be of service in effectively taking advantage of this redundancy.

III. PROPOSED APPROACH

In this paper we propose the employment of the following linear representation,

$$\Phi\{\mathbf{X}\} = \mathbf{XP} \quad (10)$$

where the matrix \mathbf{P} is a $M \times T$ matrix that causally warps all events into a common reference frame of length T (irrespective of the raw length M of \mathbf{X}). The choice of T is chosen to be larger than all training sequences. The central strength of this representation is that, depending on the nature of \mathbf{P} , *all* temporal ordering information is preserved. Further, the representation is a linear transformation of the raw signal \mathbf{X} circumventing the sometimes costly non-linear mapping required in canonical BOW representations. An obvious drawback, however, to this approach is how to obtain the alignment matrix \mathbf{P} ?

A. Choosing \mathbf{P}

An obvious choice for \mathbf{P} is simply a interpolation matrix to transform any sequence \mathbf{X} of varying frame length M into a fixed frame length sequence of T frames. This warping results in a homogeneous temporal stretching or squeezing. We shall herein refer to this interpolation matrix as $\mathbf{P}_{M \times T}^*$, which stretches a sequence of length M to length T . In all our work, we employ a linear interpolation although other interpolation strategies can be entertained.

A drawback to this naive strategy, however, is that it is almost always sub-optimal if one entertains the DTW set \mathbb{P} of all causal deformation matrices discussed in Section II-B.

For example, one can nearly always find a superior alignment between two sequences $\mathbf{X} \in \mathbb{R}^{D \times M}$ and $\mathbf{Y} \in \mathbb{R}^{D \times N}$, in terms of their Frobenius norms, such that

$$\|\mathbf{X}_i \mathbf{P}_{M \times T}^* - \mathbf{X}_j \mathbf{P}_{N \times T}^*\|_F^2 \geq \min_{\mathbf{P}_x, \mathbf{P}_y \in \mathbb{P}} \|\mathbf{X} \mathbf{P}_x - \mathbf{Y} \mathbf{P}_y\|_F^2 \quad (11)$$

where \mathbb{P} is the set of causal DTW matrices previously defined in Equation 7. An issue, however, is that the notion of alignment in Equation 11 is relative to \mathbf{X} and \mathbf{Y} . It is difficult to ascertain what \mathbf{P}_x or \mathbf{P}_y should be without knowing a priori what sequence or sequences you are aligning against.

B. Learning \mathbf{P}

Inspired by the work of [3], [4] we propose a variation upon our naive representation in Equation 10,

$$\begin{aligned} \Phi\{\mathbf{X}\} &= \frac{1}{|\mathbb{G}|} \sum_{\mathbf{P} \in \mathbb{G}} \mathbf{XP}_{M \times T}^* \mathbf{P} \\ &= \mathbf{XP}_{M \times T}^* \bar{\mathbf{P}} \end{aligned} \quad (12)$$

where \mathbb{G} is a set of *learned* $T \times T$ temporal deformation matrices. Instead of estimating an absolute alignment for \mathbf{X} , our representation instead takes the expectation of the uncertainty in absolute alignment encapsulated in the set \mathbb{G} . For computational efficiency, the summation of $\frac{1}{|\mathbb{G}|} \sum_{\mathbf{P} \in \mathbb{G}} \mathbf{P} = \bar{\mathbf{P}}$ can be pre-computed and $\mathbf{P}_{M \times T}^*$ is the linear interpolation matrix to ensure the raw sequence \mathbf{X} of length M is always of a fixed length T . We should note that we are not claiming $\bar{\mathbf{P}}$ itself to be a warping matrix (since it is the average of a set of warping matrices which belong to a non-convex set). Instead, $\bar{\mathbf{P}}$ should be just considered a pre-computation of the averaging procedure described in Equation 12.

Learning \mathbb{G} : We apply a simple but effective strategy for learning the set \mathbb{G} where we estimate the deformation matrices through the DTW objective of Equation 7 for all pairs of positive class sequence examples which we shall, for convenience, simply refer to as $\mathbf{X} \in \mathbb{R}^{D \times M}$ and $\mathbf{Y} \in \mathbb{R}^{D \times N}$. Each pair of sequences shall produce the $M \times T$ and $N \times T$ alignment matrices \mathbf{P}_x and \mathbf{P}_y respectively (they are estimated in reverse pairing as well). All these alignment matrices are collated into the learned set

$$\mathbb{G} = \{\mathbf{P}_l \mathbf{P}_{T_l \times T_{\max}}^*\}_{l=1}^L \quad (13)$$

where L is the total number of estimated deformation matrices \mathbf{P}_l across all pairs of positive class sequences, $\mathbf{P}_{T_l \times T_{\max}}^*$ is the linear interpolation matrix to scale all deformation matrices to a common length where $T_{\max} = \max\{T_l\}_{l=1}^L$ is chosen to ensure that no temporal detail is lost. Figure 1 shows four aligned pairs of videos and their corresponding warping matrix \mathbf{P}_l .

Computational Cost: Unlike the time series kernel method of [4] (see Section II-C) our proposed approach is computationally efficient. Although we cannot take advantage of the sum area table method of BOW representations, our linear approach does not require any non-linear mappings. Further, during evaluation one can actually pre-compute the application of the warping matrices,

$$\begin{aligned} f(\mathbf{X}; \boldsymbol{\theta}) &= \text{vec}\{\mathbf{XP}_{M \times T}^* \bar{\mathbf{P}}\}^T \boldsymbol{\theta} \\ &= \text{vec}\{\mathbf{X}\} \boldsymbol{\theta}_M, \end{aligned} \quad (14)$$

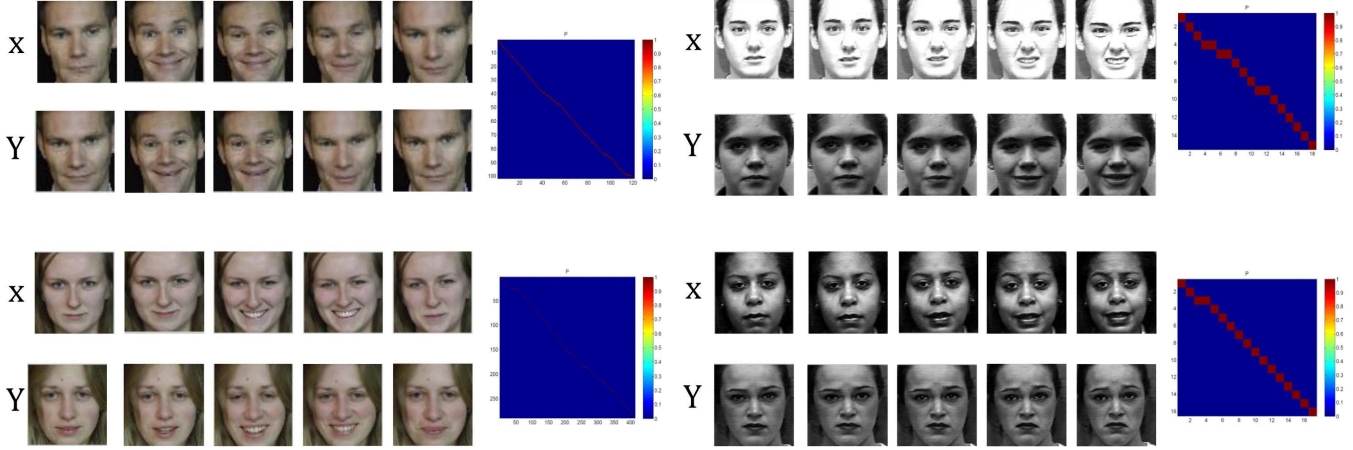


Fig. 1: Example of four aligned sequences from two different databases and their corresponding alignment paths. Figure (a)-top shows two aligned deliberate smile video frames and Figure(a)-bottom shows two aligned spontaneous smile video frames from UvA-NEMO database. Figure (b)-top shows two aligned AU-1 video frames and Figure (b)-bottom shows two aligned AU-12 video frames from CK+ database.

so that a number of $\theta_M \in \mathbb{R}^{DM \times 1}$ linear models of varying window size, M , can be pre-computed from θ so as to efficiently handle varying window sizes efficiently. The cost of evaluating a single window is then $\mathcal{O}(MD)$ which is comparable to the cost of $\mathcal{O}(KD)$ of applying the K entry codebook encoding to a new frame with a BOW representation. Also for faster detection we only select those values of M which are more likely to happen. Finally, for offline or buffered applications this approach can also utilize efficient FFT based convolutions in time to further decrease computational load.

Continuous Event Detection: Algorithm 1 shows our proposed model for detecting events in continuous video, ie. detecting a particular event in an unknown sequence with unknown starting and ending locations. We learn our model for the continuous problem by using a structured output SVM (SO-SVM) as presented in [7], because of its strengths in continuous domains. For our SO-SVM we use the same model as presented in [8], [9] for the loss function and the training model.

Non-Linear Extensions: It becomes obvious that one can apply similar a strategy for learning \mathbb{G} to the non-linear representation, $\eta\{\mathbf{X}\}$, of the codebook encoding function described in Equation 3. The only additional computational cost in testing is the $\mathcal{O}(KD)$ cost of applying the K entry codebook encoding to a new frame.

IV. FEATURE EXTRACTION FROM VIDEO

A. Feature Extraction

There are two general approaches for video feature extraction, shape-based [14], [15] and appearance-based [16], [17] methods. Common to all appearance-based methods, they have some limitations due to changes in camera view, illumination

Algorithm 1: Our Approach (Continuous Event Detection)

Input : Input examples $\mathbf{X} \in \mathbb{R}^{D \times M}$, Model parameter θ , Event size M .

Output : Event Start, Event End

Initialize: \mathbf{X}, θ, M

```

1 while  $j \in M$  do
2    $\hat{\theta}_j \leftarrow \text{linearly interpolate } \{\theta\}$ 
3    $\text{score} \leftarrow \text{conv}(\mathbf{X}, \hat{\theta}_j)$ 
4 end
5  $\{\text{start}, \text{end}\} \leftarrow \max(\text{score})$ 

```

variations, and the speed of action. On the other hand, geometric approaches follow the movement of some key parts or points (for instance on a body or face) and try to capture the temporal movement as a sequence of observations. In this paper, we use shape to represent each video frame vector. We use facial feature points and 6D comprehensive motion data, including position, orientation, acceleration and angular speed tracking for body gestures to build the observation data. The facial points are tracked using Constrained Local Models (CLM) [18]. After the facial components have been tracked, a similarity transformation is applied to facial features with respect to the normal facial shape to eliminate all variations including, scale, rotation and translation. Figure 2-b shows an example of facial landmark features in several frames of the UvA-NEMO [19] video database.

B. Feature Encoding

Shape features, \mathbf{X} , are extracted from each frame as described in Section IV-A, and are encoded in one of three ways.

Linear: refers to the raw feature representation, i.e. \mathbf{X} is used without any encoding.

Delta: refers to using a differential signal such that feature becomes $\mathbf{X}(n) - \mathbf{X}(n - 1)$.

Non-Linear: refers to the raw representation being encoded using a codebook function, $\eta\{\mathbf{X}\}$. We can also encode the delta signal with the codebook function.

V. EVALUATION

This section describes our experiments on three publicly available databases, CK+ [20] UvA-NEMO [19] and 6D Motion Gesture Database [21]. We evaluate our proposed approach for the detection of both isolated and continuous events. An overview of the databases is presented in Section V-A; Section V-B details the experimental settings used; Section V-C outlines the metrics we use to evaluate our approach; Section V-D presents our results for isolated and continuous event detection tasks; and Section V-E compares our proposed approach with other state of the art methods.

A. Databases

6D Motion Gesture Database: The 6DMG database contains comprehensive motion data, including the 3D position, orientation, acceleration, and angular speed for sets of different motion gestures performed by different users. The database contains three subsets: motion gestures, air-handwriting and air-fingerwriting. In this work we used the air-handwriting set. The WorldsViz PPT-X4 optical tracking system was used to track infra-red dots that were mounted at the top of a Wiimote. Overall, the tracking device provided 6D spatio-temporal information, including the position, orientation, acceleration and angular speed. They adjusted the scale of the 3D model to make the rendered motion as close to the real-world action as possible. This database contains 26 upper-case letters (A to Z) for motion characters. Each character is repeated 10 times for every subject. Sequences vary in duration between 27 and 412 frames. To eliminate allographs or different stroke orders, the subjects were instructed to follow a certain “stroke order” for each character (as is shown Figure 2-a).

UvA-NEMO Database: The UvA-NEMO database is collected to analyse smiles. This database is composed of video recorded with a Panasonic HDC-HS700 3MOS camcorder placed approximately 1.5 meters away from subjects. The database has 1240 smile videos in two classes, spontaneous and posed (597 spontaneous and 643 posed) from 400 subjects (185 female and 215 male). The age of subjects varies from 8 to 76 years. For posed smiles, each subject was asked to pose a smile as realistically as possible. For spontaneous smiles a short funny video was shown to each person to elicit spontaneous smiles. Each sequence starts and ends in neutral or near neutral expressions. Sequences vary in duration between 50 and 715 frames. To track the facial landmarks, we use the recently proposed CML method [18] to track 66 landmarks from each face. All tracked facial feature points are registered to a reference face by using a similarity transformation. Some examples from this database are shown in Figure 2-b.

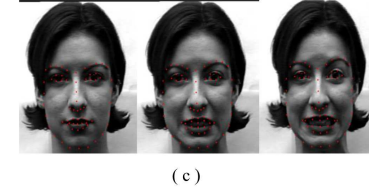
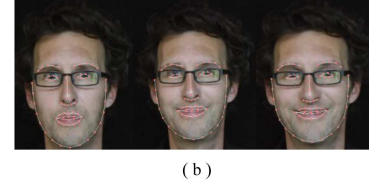
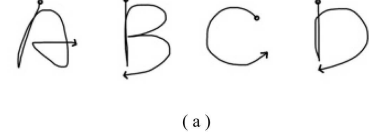


Fig. 2: a) Example of “stroke order” for 6DMG database. b) Some examples for UvA-NEMO database, c) Some examples for CK+ databases.

CK+ Database: The CK+ Database is a facial expression database. It contains 593 facial expression sequences from 123 participants. Each sequence starts from a neutral face and ends at the peak frame. Sequences vary in duration between 4 and 71 frames, and the location of 68 facial landmarks are provided along with database. Facial poses are frontal with slight head motions. All the facial feature points are registered to a reference face by using a similarity transformation. Examples from this database are shown in Figure 2-c.

B. Experimental setup

Training/Testing split: In our experiments, we use a 5-fold cross-validation to evaluate our approach. Approximately 80% of instances in each database are used for training and the remaining 20% are used for testing.

Isolated event detection task: In this task we choose three common databases 6DMG, UvA-NEMO and CK+ as presented in Subsection V-A. For each sequence the start and end points of the event of interest are known a priori. For evaluation, we use a linear SVM and LIBSVM [22] package. We perform a standard grid-search on cross-validation to tune parameters (including the C on the SVM).

Continuous event detection task: To test our proposed method on the continuous problem we use the 6DMG database. We consider detecting “A” in a word which is preceded and followed by five random letters, “B” to “Z”. The size of the sequences vary from 981 to 1482 frames. In this case the start and end points of the event of interest are unknown. We use the 6DMG database for the continuous event detection problem because it has longer videos compared to other databases. For

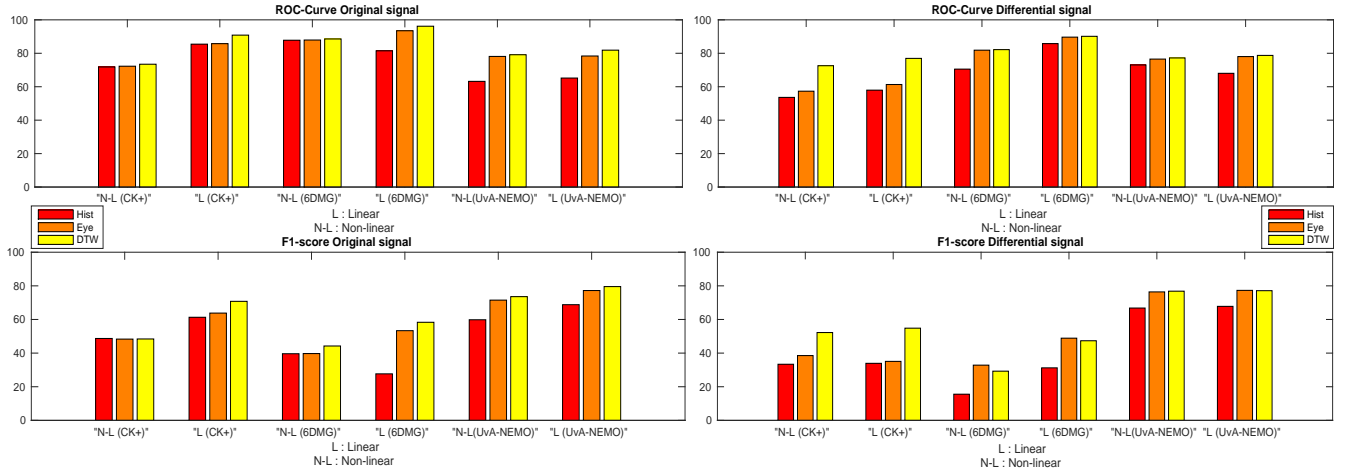


Fig. 3: Graphs comparing the accuracy and F_1 -score on Non-Linear and Linear features with different $\bar{\mathbf{P}}$ models. Delta corresponds to the differential signal ($\mathbf{X}(n) - \mathbf{X}(n-1)$).

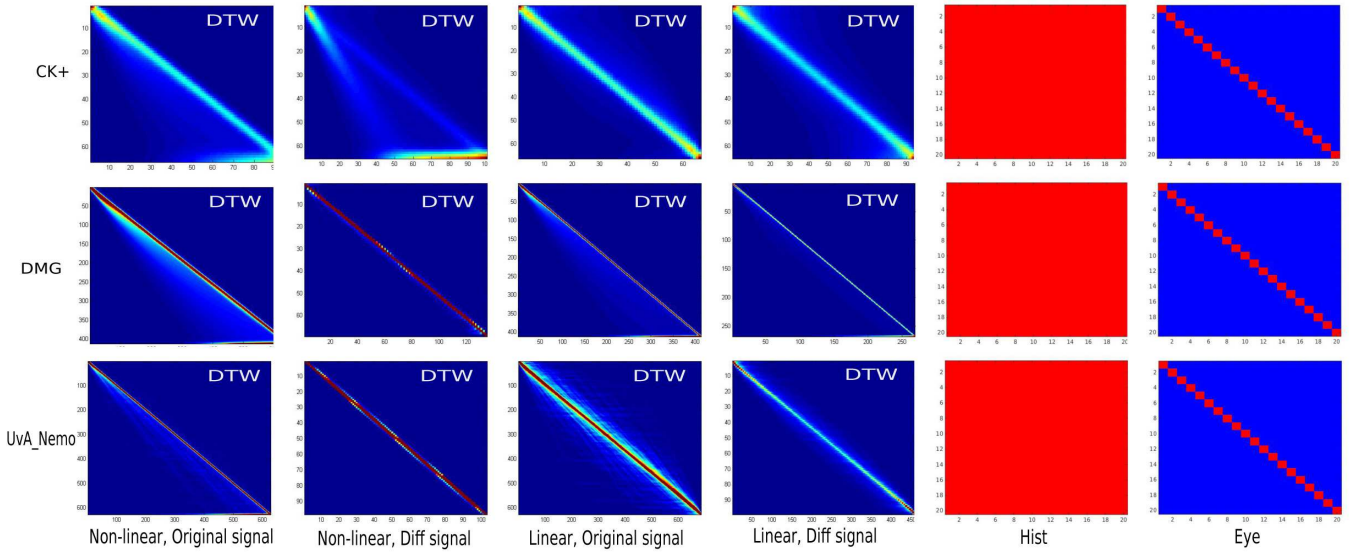


Fig. 4: The mapping matrix $\bar{\mathbf{P}}$ for seven different cases across the three databases (6DMG, CK+ and Uva-NEMO): The first four columns show the proposed approach with different feature encodings (see Section IV-B); the fifth column shows $\bar{\mathbf{P}}$ for the Histogram case; and the last column shows $\bar{\mathbf{P}}$ for the Identity case.

evaluation, we use SO-SVM (using the $\text{SVM}^{\text{struct}}$ package ¹). We perform a standard grid-search on the validation set to tune parameters (including parameter C in SO-SVM).

Number of temporal codebooks: For building the codebooks, k -means clustering is used. In our experiments we perform cross-validation to tune the number of temporal codebooks. In this work we set 300 codebooks for 6DMG, 136 for CK+ and 1500 for the UvA-NEMO database for the original signal, and in the case of delta signal we set these values to 100, 30, and 500 respectively.

C. Evaluation metrics

To evaluate the performance, we report the area under ROC curve, and the maximum F_1 -score. The F_1 -score is defined as: $F_1 = \frac{2 \times \text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}}$, and conveys the balance between the precision and recall. The F_1 -score is a better performance measure than the area under ROC curve because the ROC curve is designed to measure the binary classification rather than detection and fails to reflect the effect of the proportion of the positive to negative samples.

D. Results

Figure 3 compares the performance of different configurations of the proposed approach, reporting the average accuracy

¹available at: http://www.cs.cornell.edu/people/tj/svm_light/svm_struct.html

Method	Area under ROC curve	
	Computational time (s)	
BOW + SO-SVM [8], [9]	135.8995	56.27
Our method + SO-SVM	81.3082	58.30

TABLE I: Comparing our proposed approach (using a linear encoding of the original signal) with methods of [8], [9] and [4] on three databases. The table shows the area under ROC curve.

Method	Area under ROC curve		
	CK+	6DMG	UvA-NEMO
BOW + SVM [8], [9]	71.83	87.81	63.21
Lőrincz et al. [4]	89.13	89.77	75.25
Our method + SVM	90.86	96.19	81.87

TABLE II: Comparing our proposed approach (using a linear encoding of the original signal) with methods of [8], [9] and [4] on three databases. The table shows the area under ROC curve.

and F_1 -score among all classes, and Figure 4 shows the variations of $\bar{\mathbf{P}}$ used in Equation 12 learned using the proposed approach in Section III-B. We investigate the impact of different feature encodings (see Section IV-B): “Linear” refers to using the raw representation \mathbf{X} ; “Delta” refers to using differential signal $\mathbf{X}(n) - \mathbf{X}(n-1)$; and “Non-Linear” refers to using the codebook encoding function $\eta\{\mathbf{X}\}$, which we also apply to the differential signal. Figure 4 visualises the $\bar{\mathbf{P}}$ matrices learned through the DTW procedure described in Section III-B, and we also compare to two other representations: “Hist”, where all elements of $\bar{\mathbf{P}}$ are set to unity; and “Eye”, where $\bar{\mathbf{P}}$ is simply an identity matrix.

It is interesting to note that our representation, when employing HIST for $\bar{\mathbf{P}}$ in conjunction with a Non-Linear representation $\eta\{\mathbf{X}\}$, is equivalent to the BOW representation described in Equation 2. We can see that using the non-linear representation with a histogram for $\bar{\mathbf{P}}$ (i.e. BOW), performs poorly. This is to be expected as the BOW representation throws away all temporal information. On the other hand, stretching the observations to a standard length (linear interpolation, “Eye”) shows better performance than using a histogram, as this preserves some temporal ordering. As can be seen from the graphs, our proposed model outperforms both the BOW and naive interpolation methods. In this case learning $\bar{\mathbf{P}}$ from DTW alignment helps the model to preserve the temporal ordering information. The results also show that using the non-linear representation degrades performance across all datasets and types of $\bar{\mathbf{P}}$ matrix.

Table ?? shows performance for the continuous event detection problem, and compares the run times and area under ROC curve of our proposed method (using a linear encoding of the original signal) from Section III-B with that of BOW method. The cost of search in our proposed model is much less than using BOW, while also achieving better performance. The run times shown in Table ?? are achieved using Matlab implementations on a Intel i7 2.1GHZ dual core CPU with 16GB RAM.

Method	F_1 -score		
	CK+	6DMG	UvA-NEMO
BOW + SVM [8], [9]	48.70	39.64	59.84
Lőrincz et al. [4]	71.33	53.84	78.50
Our method + SVM	70.79	58.33	79.56

TABLE III: Comparing our proposed approach (using a linear encoding of the original signal) with methods of [8], [9] and [4] on three databases. The table shows the F_1 -score.

E. Comparing with other methods

In this subsection we compare our method (using a linear encoding of the original signal) with the state-of-the-art BOW method [8], [9] and the time series kernel method of Lőrincz et al. [4].

The BOW method was proposed to tackle the problem of action unit detection. [8], [9] compared their method with a frame-based SVM approach and a dynamic method using HMMs. They showed a segment-based SVM classifier using BOW feature vectors outperforms both a frame-based SVM and a HMM with two or four states. The major difference between frame-based SVM and segment-based one is the former classifies each frame independently while the latter considers collection of frames for prediction. We implement segment-based SVM using BOW proposed by [8], [9] and compare it against our proposed approach introduced in Section III-B. The area under ROC curve and F_1 -score for this comparison are reported in Table II and Table III on above-mentioned databases. As shown, our approach significantly outperforms segment-based SVM.

We also compare our method against Lőrincz et al. [4]. They proposed to use a time series kernel for event detection and obtained state-of-the-art performance for expression classification. As can be seen, our method outperforms [4]. We also note that the computational cost for our proposed method is $\mathcal{O}(MD)$, however the computational complexity of using time series kernel in [4] is $\mathcal{O}(LMND)$ where L is the number of support vectors, M is the length of the input sequence, N is the average length of the support vector sequences and D is the dimensionality of the sequences.

VI. CONCLUSION

In this paper we addressed the problem of event detection and presented a simple, yet efficient, approach. Our proposed algorithm preserves temporal ordering that is essential for the analysis of problems with a dynamic nature. In this approach, instead of aligning all sequences to a single temporal reference, we employed the notion of relative alignment between pairs of training examples. This approach proved effective in our empirical evaluations and maintained ordering whilst preserving the discriminative characteristics of the problem. We also demonstrated how the proposed approach could be extended to tackle the problem of continuous event detection, and demonstrated efficient and accurate performance.

ACKNOWLEDGMENT

This research was supported by Australian Research Council (ARC) Discovery Grant DP140100793.

REFERENCES

- [1] L. Rabiner and B.-H. Juang, "An introduction to hidden markov models," *IEEE ASSP Magazine*, vol. 3, no. 1, pp. 4–16, 1986.
- [2] L. Morency, A. Quattoni, and T. Darrell, "Latent-dynamic discriminative models for continuous gesture recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2007, pp. 1–8.
- [3] M. Cuturi, "Fast global alignment kernels," in *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, 2011, pp. 929–936.
- [4] A. Lorincz, L. Attila Jeni, Z. Szabo, J. F. Cohn, and T. Kanade, "Emotional expression classification using time-series kernels," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2013.
- [5] H. S. K.-i. Noma, "Dynamic time-alignment kernel in support vector machine," *Advances in neural information processing systems*, vol. 14, p. 921, 2002.
- [6] V. N. Vapnik and V. Vapnik, *Statistical learning theory*. Wiley New York, 1998, vol. 2.
- [7] I. Tsochantaris, T. Joachims, T. Hofmann, and Y. Altun, "Large margin methods for structured and interdependent output variables," in *Journal of Machine Learning Research*, 2005, pp. 1453–1484.
- [8] M. Hoai and F. de la Torre, "14 structured prediction for event detection," *Advanced Structured Prediction*, p. 333, 2014.
- [9] M. Hoai and F. De la Torre, "Max-margin early event detectors," *International Journal of Computer Vision*, vol. 107, no. 2, pp. 191–202, 2014.
- [10] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1. IEEE, 2001, pp. 1–511.
- [11] S. A. Kurtsek, A. Srivastava, and W. Wu, "Signal estimation under random time-warps and nonlinear signal alignment," in *Advances in Neural Information Processing Systems*, 2011, pp. 675–683.
- [12] F. Zhou and F. Torre, "Canonical time warping for alignment of human behavior," in *Advances in neural information processing systems*, 2009, pp. 2286–2294.
- [13] L. R. Rabiner and B.-H. Juang, *Fundamentals of speech recognition*. PTR Prentice Hall Englewood Cliffs, 1993, vol. 14.
- [14] S. Carlsson and J. Sullivan, "Action recognition by shape matching to key frames," in *Workshop on Models versus Exemplars in Computer Vision*, vol. 1, 2001, p. 18.
- [15] M. F. Valstar, I. Patras, and M. Pantic, "Facial action unit detection using probabilistic actively learned support vector machines on tracked facial point data," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*. IEEE, 2005, pp. 76–76.
- [16] Y. Zhu, F. De la Torre, J. F. Cohn, and Y.-J. Zhang, "Dynamic cascades with bidirectional bootstrapping for spontaneous facial action unit detection," in *International Conference on Affective Computing and Intelligent Interaction and Workshops*. IEEE, 2009, pp. 1–8.
- [17] K. Sikka, T. Wu, J. Susskind, and M. Bartlett, "Exploring bag of words architectures in the facial expression domain," in *ECCV Workshops and Demonstrations*. Springer, 2012, pp. 250–259.
- [18] A. Asthana, S. Zafeiriou, S. Cheng, and M. Pantic, "Robust discriminative response map fitting with constrained local models," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2013, pp. 3444–3451.
- [19] H. Dibeklioglu, A. A. Salah, and T. Gevers, "Are you really smiling at me? spontaneous versus posed enjoyment smiles," in *European Conference on Computer Vision*. Springer, 2012, pp. 525–538.
- [20] P. Lucey, J. F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, and I. Matthews, "The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. IEEE, 2010, pp. 94–101.
- [21] M. Chen, G. AlRegib, and B.-H. Juang, "6dmg: a new 6d motion gesture database," in *Proceedings of the 3rd Multimedia Systems Conference*. ACM, 2012, pp. 83–88.
- [22] C.-C. Chang and C.-J. Lin, "Libsvm: a library for support vector machines," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 2, no. 3, p. 27, 2011.